UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/806,871	03/22/2004	Robert Tod Dimpsey	AUS920040064US1	2672

35525 7590 10/31/2008

IBM CORP (YA)
C/O YEE & ASSOCIATES PC
P.O. BOX 802333
DALLAS, TX 75380

EXAMINER	
SAVLA, ARPAN P	

ART UNIT	PAPER NUMBER
2185	

NOTIFICATION DATE	DELIVERY MODE
10/31/2008	ELECTRONIC

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Notice of the Office communication was sent electronically on above-indicated "Notification Date" to the following e-mail address(es):

ptonotifs@yeeiplaw.com

UNITED STATES PATENT AND TRADEMARK OFFICE

Commissioner for Patents
United States Patent and Trademark Office
P.O. Box 1450
Alexandria, VA 22313-1450
www.uspto.gov

BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

Application Number: 10/806,871
Filing Date: March 22, 2004
Appellant(s): DIMPSEY ET AL.

Gerald H. Glanzman
For Appellant

EXAMINER'S ANSWER

This is in response to the appeal brief filed August 11, 2008 appealing from the Office action mailed April 9, 2008.

(1) Real Party in Interest

A statement identifying by name the real party in interest is contained in the brief.

(2) Related Appeals and Interferences

The examiner is not aware of any related appeals, interferences, or judicial proceedings which will directly affect or be directly affected by or have a bearing on the Board's decision in the pending appeal.

(3) Status of Claims

The statement of the status of claims contained in the brief is correct.

(4) Status of Amendments After Final

The appellant's statement of the status of amendments after final rejection contained in the brief is correct.

(5) Summary of Claimed Subject Matter

The summary of claimed subject matter contained in the brief is correct.

(6) Grounds of Rejection to be Reviewed on Appeal

The appellant's statement of the grounds of rejection to be reviewed on appeal is correct.

(7) Claims Appendix

The copy of the appealed claims contained in the Appendix to the brief is correct.

(8) Evidence Relied Upon

6,381,679	**Matsubara et al.**	**4-2002**
5,708,803	**Ishimi et al.**	**1-1998**
2003/0191900	**Hooker**	**10-2003**

Anonymously Disclosed, "Method for the dynamic prediction of nonsequential memory accesses", September 25, 2002, ip.com, IPCOM000009888D

(9) Grounds of Rejection

The following ground(s) of rejection are applicable to the appealed claims:

Claims 1-2, 5-7, 11-12, 15-19, and 22-24 are rejected under 35 U.S.C. 103(a) as being obvious over Matsubara et al. (U.S. Patent 6,381,679) (hereinafter "Matsubara") in view of Anonymously Disclosed, "Method for the dynamic prediction of nonsequential memory accesses" (hereinafter "Anon") and Ishimi et al. (U.S. Patent 5,708,803) (hereinafter "Ishimi").

As per claims 1 and 18, Matsubara discloses a method in a data processing system for providing hardware assistance to prefetch data during execution of code by a

processor in the data processing system, the method comprising:

responsive to loading an instruction in the code into a cache, determining, by a processor unit, whether a prefetch indicator is associated with the instruction (col. 5, lines 1-10; col. 6, lines 35-42 and 53-55; col. 7, lines 10-20; Fig. 1; Fig. 2, elements 21 and 22; Fig. 6B). *It should be noted that computer program product in claims 18-19 and 21-24 executes the exact same functions as the methods in claims 1-2 and 4-7. Therefore, any references that teach claims 1-2 and 4-7 also teach the corresponding claims 18-19 and 21-24. It should also be noted that the "indication bits (i.e. PF bits)" equaling 1 is analogous to the "prefetch indicator being associated with the instruction" and the "CPU 21" is analogous to the "processor unit." Lastly, it should be noted that the "instruction fetch (IF)" stage is when the instruction in the code is loaded into a cache and the "decoding" stage is when the "determination" is made.*

and responsive to the prefetch indicator being associated with the instruction, selectively prefetching data into the cache in the processor (col. 6, lines 53-55; Fig. 2, elements 21 and 22). *It should be noted that when it is determined that the value of the PF bits is 1, all the data of the line is prefetched to the primary cache.*

Matsubara does not expressly disclose a pointer to a data structure identified by the prefetch indicator;

wherein the selectively prefetching step includes:

determining whether outstanding cache misses are present;

and prefetching the data in response to a determination that a number of outstanding cache misses is less than a threshold.

Anon discloses a pointer to a data structure identified by the prefetch indicator (General Description, 1st paragraph and 4th paragraph; Detailed Description, 1st paragraph). *It should be noted that the "dynamic prefetch pointer" is analogous to the "pointer to a data structure."*

Matsubara and Anon are analogous art because they are from the same field of endeavor, that being prefetching memory systems.

At the time of the invention it would have been obvious to a person of ordinary skill in the art to implement Anon's dynamic prefetch pointer within Matsubara's information processing system.

The motivation for doing so would have been to improve memory access due to improved memory access prediction and also improve performance due to reducing the time spent waiting for memory accesses to complete (Anon, General Description, 5th paragraph).

The combination of Matsubara/Anon does not expressly disclose wherein the selectively prefetching step includes:

determining whether outstanding cache misses are present;

and prefetching the data in response to a determination that a number of outstanding cache misses is less than a threshold.

Ishimi discloses wherein the selectively prefetching step includes:

determining whether outstanding cache misses are present (col. 13, line 30; Fig. 13, element S4).

and prefetching the data in response to a determination that a number of outstanding cache misses is less than a threshold (col. 13, lines 30-32; Fig. 13, element S10). *It should be noted that the threshold is equal to 1. Thus, when it is determined there is a cache hit, meaning there are zero outstanding cache misses (i.e. the number of outstanding cache misses is than the threshold of 1), data is prefetched.*

The combination of Matsubara/Anon and Ishimi are analogous art because they are from the same field of endeavor, that being prefetching memory systems.

At the time of the invention it would have been obvious to a person of ordinary skill in the art to implement Ishimi's fetch mechanism within Matsubara/Anon's information processing system.

The motivation for doing so would have been to provide a data processor capable of processing quickly by lessening the number of abortions even when a branch prediction is preformed (Ishimi, col. 3, lines 3-6).

Therefore, it would have been obvious to combine Matsubara, Anon, and Ishimi for the benefit of obtaining the invention as specified in claims 1 and 18.

As per claims 2 and 19, the combination of Matsubara/Anon/Ishimi discloses the prefetch indicator contains the pointer to the data structure (Anon, General Description, 4th paragraph).

As per claims 5 and 22, the combination of Matsubara/Anon/Ishimi discloses the processor unit is selected from one of an instruction cache, data cache, and a

load/store unit (Matsubara, col. 6, lines 35-42; col. 7, lines 10-20; Fig. 2, element 21). *It should be noted that the "CPU 21" is analogous to a "load/store unit."*

As per claims 6 and 23, the combination of Matsubara/Anon/Ishimi discloses the cache is an instruction cache (Ishimi, col. 1, lines 25-28).

As per claims 7 and 24, the combination of Matsubara/Anon/Ishimi discloses the cache is a data cache (Matsubara, col. 8, lines 56-63).

As per claim 11, Matsubara discloses a data processing system for providing hardware assistance to prefetch data during execution of code by a processor in the data processing system, the data processing system comprising:

determining means, responsive to loading an instruction in the code into a cache, for determining, by the a processor unit, whether a prefetch indicator is associated with the instruction (col. 5, lines 1-10; col. 6, lines 35-42 and 53-55; col. 7, lines 10-20; Fig. 1; Fig. 2, elements 21 and 22; Fig. 6B); *It should be noted that pg. 13, lines 3-5 of Applicant's specification appear to define this means as a computer. Also, see the citation note for the similar limitation in claims 1 and 18 above.*

and selectively prefetching means, responsive to the prefetch indicator being associated with the instruction, for selectively prefetching data into the cache in the processor (col. 6, lines 53-55; Fig. 2, elements 21 and 22). *It should be noted that pg.*

13, lines 3-5 of Applicant's specification appear to define this means as a computer. Also, see the citation note for the similar limitation in claims 1 and 18 above.

Matsubara does not expressly disclose a pointer to a data structure identified by the prefetch indicator;

wherein the selectively prefetching means includes:

means for determining whether outstanding cache misses are present;

and means for prefetching the data in response to a determination that a number of outstanding cache misses is less than a threshold.

Anon discloses a pointer to a data structure identified by the prefetch indicator (General Description, 1st paragraph and 4th paragraph; Detailed Description, 1st paragraph). *See the citation note for the similar limitation in claims 1 and 18 above.*

Matsubara and Anon are analogous art because they are from the same field of endeavor, that being prefetching memory systems.

At the time of the invention it would have been obvious to a person of ordinary skill in the art to implement Anon's dynamic prefetch pointer within Matsubara's information processing system.

The motivation for doing so would have been to improve memory access due to improved memory access prediction and also improve performance due to reducing the time spent waiting for memory accesses to complete (Anon, General Description, 5th paragraph).

The combination of Matsubara/Anon does not expressly disclose wherein the selectively prefetching means includes:

means for determining whether outstanding cache misses are present;

and means for prefetching the data in response to a determination that a number of outstanding cache misses is less than a threshold.

Ishimi discloses wherein the selectively prefetching means includes:

means for determining whether outstanding cache misses are present (col. 13, line 30; Fig. 13, element S4).

and means for prefetching the data in response to a determination that a number of outstanding cache misses is less than a threshold (col. 13, lines 30-32; Fig. 13, element S10). *See the citation note for the similar limitation in claims 1 and 18 above.*

The combination of Matsubara/Anon and Ishimi are analogous art because they are from the same field of endeavor, that being prefetching memory systems.

At the time of the invention it would have been obvious to a person of ordinary skill in the art to implement Ishimi's fetch mechanism within Matsubara/Anon's information processing system.

The motivation for doing so would have been to provide a data processor capable of processing quickly by lessening the number of abortions even when a branch prediction is preformed (Ishimi, col. 3, lines 3-6).

Therefore, it would have been obvious to combine Matsubara, Anon, and Ishimi for the benefit of obtaining the invention as specified in claim 11.

As per claim 12, the combination of Matsubara/Anon/Ishimi discloses the prefetch indicator contains the pointer to the data structure (Anon, General Description,

4th paragraph).

As per claim 15, the combination of Matsubara/Anon/Ishimi discloses the processor unit is selected from one of an instruction cache, data cache, and a load/store unit (Matsubara, col. 6, lines 35-42; col. 7, lines 10-20; Fig. 2, element 21). *See the citation note for claims 5 and 22 above.*

As per claim 16, the combination of Matsubara/Anon/Ishimi discloses the cache is an instruction cache (Ishimi, col. 1, lines 25-28).

As per claim 17, the combination of Matsubara/Anon/Ishimi discloses the cache is a data cache (Matsubara, col. 8, lines 56-63).

Claims 4, 14, and 21 are rejected under 35 U.S.C. 103(a) as being obvious over Matsubara in view of Anon and Ishimi as applied to claims 1, 11, and 18 above, and further in view of Hooker (U.S. Patent Application Publication 2003/0191900).

As per claims 4 and 21, the combination of Matsubara/Anon/Ishimi discloses all the limitations of claims 4 and 21 except wherein the selectively prefetching step further includes:

determining whether to replace cache lines;

and prefetching the in response to a determination that a number of cache lines chosen to replaced is greater than a threshold.

Hooker discloses wherein the selectively prefetching step further includes:

determining whether to replace cache lines (paragraph 0069; Fig. 5, element 536); *It should be noted that the "response buffers" are analogous to the "cache lines."*

and prefetching the in response to a determination that a number of cache lines chosen to replaced is greater than a threshold (paragraph 0070; Fig. 5, element 538).

The combination of Matsubara/Anon/Ishimi and Hooker are analogous art because they are from the same field of endeavor, that being prefetching memory systems.

At the time of the invention it would have been obvious to a person of ordinary skill in the art to implement Hooker's threshold based prefetch method into Matsubara/Anon/Ishimi's information processing system.

The motivation for doing so would have been to reduce software code size over conventional single-cache line prefetch instructions because fewer prefetch instructions need to be included in the program (Hooker, paragraph 0018). Another motivation for doing so would have been to potentially improve system performance by making more efficient use of the processor bus than the conventional method (Hooker, paragraph 0018). Lastly, another motivation for doing so would have been to potentially improve processing performance by moving data into the microprocessor cache more efficiently than the conventional method by alleviating the problems caused by the fact that a range of core clock to processor bus clock ratios may exist (Hooker, paragraph 0018).

Therefore, it would have been obvious to combine Matsubara, Anon, Ishimi, and Hooker for the benefit of obtaining the invention as specified in claims 4 and 21.

As per claim 14, the combination of Matsubara/Anon/Ishimi/Hooker discloses wherein the selectively prefetching means further includes:

means for determining whether to replace cache lines (paragraph 0069; Fig. 5, element 536); *See the citation note for claims 4 and 21 above.*

and means for prefetching the in response to a determination that a number of cache lines chosen to replaced is greater than a threshold (paragraph 0070; Fig. 5, element 538).

Claim 25 is rejected under 35 U.S.C. 103(a) as being obvious over Matsubara in view of Anon, Ishimi, and Hooker.

As per claim 25, Matsubara discloses a method in a data processing system for providing hardware assistance to prefetch data during execution of code by a processor in the data processing system, the method comprising:

responsive to loading an instruction in the code into a cache, determining, by a processor unit, whether a prefetch indicator is associated with the instruction (col. 5, lines 1-10; col. 6, lines 35-42 and 53-55; col. 7, lines 10-20; Fig. 1; Fig. 2, elements 21 and 22; Fig. 6B), wherein the processor unit is selected from one of an instruction cache, data cache, or a load/store unit (Matsubara, col. 6, lines 35-42; col. 7, lines 10-20; Fig. 2, element 21).

and responsive to the prefetch indicator being associated with the instruction, selectively prefetching data into the cache in the processor (col. 6, lines 53-55; Fig. 2, elements 21 and 22).

Matsubara does not expressly disclose a pointer to a data structure identified by the prefetch indicator;

wherein the selectively prefetching step includes:

determining whether outstanding cache misses are present;

and prefetching the data in response to a determination that a number of outstanding cache misses is less than a threshold, and wherein the selectively prefetching step further includes:

determining whether to replace cache lines;

and prefetching the in response to a determination that a number of cache lines chosen to replaced is greater than a threshold.

Anon discloses a pointer to a data structure identified by the prefetch indicator (General Description, 1st paragraph and 4th paragraph; Detailed Description, 1st paragraph). *See the citation notes from the similar limitations in claims 1 and 18 above.*

Matsubara and Anon are analogous art because they are from the same field of endeavor, that being prefetching memory systems.

At the time of the invention it would have been obvious to a person of ordinary skill in the art to implement Anon's dynamic prefetch pointer within Matsubara's information processing system.

The motivation for doing so would have been to improve memory access due to improved memory access prediction and also improve performance due to reducing the time spent waiting for memory accesses to complete (Anon, General Description, 5th paragraph).

The combination of Matsubara/Anon does not expressly disclose wherein the selectively prefetching step includes:

determining whether outstanding cache misses are present;

and prefetching the data in response to a determination that a number of outstanding cache misses is less than a threshold, and wherein the selectively prefetching step further includes:

determining whether to replace cache lines;

and prefetching the in response to a determination that a number of cache lines chosen to replaced is greater than a threshold.

Ishimi discloses wherein the selectively prefetching step includes:

determining whether outstanding cache misses are present (col. 13, line 30; Fig. 13, element S4).

and prefetching the data in response to a determination that a number of outstanding cache misses is less than a threshold (col. 13, lines 30-32; Fig. 13, element S10).

The combination of Matsubara/Anon and Ishimi are analogous art because they are from the same field of endeavor, that being prefetching memory systems.

At the time of the invention it would have been obvious to a person of ordinary skill in the art to implement Ishimi's fetch mechanism within Matsubara/Anon's information processing system.

The motivation for doing so would have been to provide a data processor capable of processing quickly by lessening the number of abortions even when a branch prediction is preformed (Ishimi, col. 3, lines 3-6).

The combination of Matsubara/Anon/Ishimi does not expressly disclose wherein the selectively prefetching step further includes:

determining whether to replace cache lines;

and prefetching the in response to a determination that a number of cache lines chosen to replaced is greater than a threshold.

Hooker discloses wherein the selectively prefetching step further includes:

determining whether to replace cache lines (paragraph 0069; Fig. 5, element 536);

and prefetching the in response to a determination that a number of cache lines chosen to replaced is greater than a threshold (paragraph 0070; Fig. 5, element 538).

The combination of Matsubara/Anon/Ishimi and Hooker are analogous art because they are from the same field of endeavor, that being prefetching memory systems.

At the time of the invention it would have been obvious to a person of ordinary skill in the art to implement Hooker's threshold based prefetch method into Matsubara/Anon/Ishimi's information processing system.

The motivation for doing so would have been to reduce software code size over conventional single-cache line prefetch instructions because fewer prefetch instructions need to be included in the program (Hooker, paragraph 0018). Another motivation for doing so would have been to potentially improve system performance by making more efficient use of the processor bus than the conventional method (Hooker, paragraph 0018). Lastly, another motivation for doing so would have been to potentially improve processing performance by moving data into the microprocessor cache more efficiently than the conventional method by alleviating the problems caused by the fact that a range of core clock to processor bus clock ratios may exist (Hooker, paragraph 0018).

Therefore, it would have been obvious to combine Matsubara, Anon, Ishimi, and Hooker for the benefit of obtaining the invention as specified in claim 25.

(10) Response to Arguments

Response to A.1.

Appellant argues, in section A.1. and more specifically page 13 of the Appeal Brief, that:

"Ishii (sic) does not teach or in any suggest "determining whether outstanding cache misses are present; and prefetching the data in response to a determination that a number of outstanding cache misses is less than a threshold" as recited in claim 1."

The Examiner respectfully disagrees. The Examiner submits that there is a need to look closely at the implicit disclosure Ishimi provides. It is important to remember that

in Ishimi each cache hit or miss is dealt with right after a cache access operation (see Fig. 13, S1-S4). Neither cache hits nor cache misses are allowed to accumulate. What this means is that once a cache hit or a miss occurs, appropriate steps are taken within the data processor to properly handle (and consequently dispose of) the cache hit or miss before the next cache access operation occurs (and therefore before the next cache hit or miss occurs). This can be clearly seen from the flow chart that appears in Fig. 13 of Ishimi. Thus, in Ishimi there can only be a single outstanding cache hit or equally a single outstanding cache miss at any given time. Therefore, not only does a cache hit indicate there is not a cache miss, but a cache hit also indicates that there are zero outstanding cache misses in the system. Keeping that in mind, the Examiner submits that Ishimi clearly discloses determining whether outstanding cache misses are present at col. 13, line 30 and Fig. 13, element S4. Appellant themselves admit that Ishimi, Figure 13 at S4 depicts a determination is made as to whether there is a cache hit or a cache miss (see Appeal Brief, page 13). Accordingly, Ishimi sufficiently discloses determining whether outstanding cache misses are present.

Next there will be an evaluation of Ishimi's data processor functions under different conditions. In the case where it is determined that the number of outstanding cache misses is greater than 0, meaning there is a cache miss (i.e. there is not a cache hit), the fetch operation is discontinued (see Fig. 13, S4-S5). In the case where it is determined that the number of outstanding cache misses is less than 1, meaning there are zero outstanding cache misses (i.e. there is a cache hit), data is prefetched from the cache (see Fig. 13, S4 and S10). While Ishimi may not explicitly teach an active

threshold, it is clear, based on how the data processor functions, that an implicit threshold is used in determination step S4 in Fig. 13. Consequently, Ishimi implicitly discloses Appellant's threshold, as simply and broadly claimed. Thus, in Ishimi when a determination is made that the number of outstanding cache misses is less than the implicit threshold of 1, in response to such a determination, data is prefetched from the cache. Accordingly, Ishimi sufficiently discloses prefetching the data in response to a determination that a number of outstanding cache misses is less than a threshold. Based on the foregoing, the combination of Matsubara/Anon/Ishimi renders claim 1 unpatentable.

Response to A.2.

Appellant argues, in section A.2. of the Appeal Brief, that:

"The publication does not disclose or suggest a prefetch indicator (as specifies in the independent claims) that contains a pointer to the data structure identified by the prefetch indicator."

The Examiner respectfully disagrees. It appears Appellant is attacking the references individually. One cannot show nonobviousness by attacking references individually where the rejections are based on combinations of references. See *In re Keller*, 642 F.2d 413, 208 USPQ 871 (CCPA 1981); *In re Merck & Co.*, 800 F.2d 1091, 231 USPQ 375 (Fed. Cir. 1986). The rejection of claims 2, 12, and 19 is not based on individual references, but rather the combination of Matsubara, Anon, and Ishimi. In the rejection of claims 2, 12, and 19, when combining Matsubara and Anon, as set forth by

the Examiner above, Anon's dynamic prefetch pointer is implemented within Matsubara's information processing system. In such an implementation Anon's dynamic prefetch pointer is combined with Matsubara's PF bits. In such a combination, the prefetch indicator ("PF bits") contains a pointer ("dynamic prefetch pointer") to a data structure identified by the prefetch indicator. Accordingly, the combination of Matsubara/Anon/Ishimi renders claims 2, 12, and 19 unpatentable.

Response to B.

Appellant argues, in section B. of the Appeal Brief, that:

"Hooker does not supply the deficiencies in Matsubara, Anon and Ishii described above with respect to the independent claims, nor does the Examiner assert otherwise."

The Examiner respectfully disagrees and refers Appellant to the response to A.1. above which details how Ishimi discloses determining whether outstanding cache misses are present and prefetching the data in response to a determination that a number of outstanding cache misses is less than a threshold. Accordingly, the combination of Matsubara/Anon/Ishimi/Hooker renders claims 4, 14, and 21 unpatentable.

Response to C.

Appellant argues, in section C. of the Appeal Brief, that:

"For similar reasons as described in detail in Sections A and B above, Appellants respectfully submit that the Examiner has not established a prima facie case of

obviousness in rejecting claim 25 because neither Matsubara nor Anon nor Ishimi nor Hooker nor their combination discloses or suggests "determining that outstanding cache misses are present; and prefetching the data in response to a determination that a number of outstanding cache misses is less than a threshold" as recited in claim 25."

The Examiner respectfully disagrees and refers Appellant to the response to A.1. above which details how Ishimi discloses determining whether outstanding cache misses are present and prefetching the data in response to a determination that a number of outstanding cache misses is less than a threshold. Accordingly, the combination of Matsubara/Anon/Ishimi/Hooker renders claim 25 unpatentable.

(11) Related Proceeding(s) Appendix

No decision rendered by a court or the Board is identified by the examiner in the Related Appeals and Interferences section of this examiner's answer.

For the above reasons, it is believed that the rejections should be sustained.

Respectfully submitted,

/Arpan Savla/

Examiner, Art Unit 2185

Conferees:

/Sanjiv Shah/

Supervisory Patent Examiner, Art Unit 2185

/Manorama Padmanabhan/

Quality Assurance Specialist, TC 2100, WG 2180